University of Baghdad
College of Engineering
JE
JOURNAL OF ENGINEERING

Journal of Engineering
journal homepage: www.joe.uobaghdad.edu.iq
Number 6 Volume 25 June 2019

Chemical, Petroleum and Environmental Engineering

A Comparative Study of Various Intelligent Algorithms based Path Planning for Mobile Robots

Muna Mohammed Jawad*
Assist Professor
Control and Systems Eng. Dept.
University of Technology
E-mail: Muna Al-Nayar @yahoo.com

Esraa Adnan Hadi
M.Sc Student
Control and Systems Eng. Dept.
University of Technology
E-mail: engesraa753@yahoo.com

ABSTRACT

In general, path-planning problem is one of most important task in the field of robotics. This paper describes the path-planning problem of mobile robot based on various metaheuristic algorithms. The suitable collision free path of a robot must satisfies certain optimization criteria such as feasibility, minimum path length, safety and smoothness and so on. In this research, various three approaches namely, PSO, Firefly and proposed hybrid FFCPSO are applied in static, known environment to solve the global path-planning problem in three cases. The first case used single mobile robot, the second case used three independent mobile robots and the third case applied three follow up mobile robot. Simulation results, which carried out using MATLAB 2014 environment, show the validity of the kinematic model for Nonholonomic mobile robot and demonstration that the proposed algorithm perform better than original PSO and FF algorithms under the same environmental constraints by providing the smoothness velocity and shortest path for each mobile robot.

Keywords: wheeled mobile robot, path planning, static environment, firefly algorithm, particle swarm optimization algorithm.

دراسة مقارنة لخوارزميات ذكية مختلفة القائمة على تخطيط المسارات لعدد من الروبوتات المتنقلة

الأستاذ المساعد الدكتورة منى محمد جواد
قسم هندسة السيطرة والنظم
الجامعة التكنولوجية

أسراء عدنان هادي
قسم هندسة السيطرة والنظم
الجامعة التكنولوجية

الخلاصة

بشكل عام تعتبر مشكلة تخطيط المسار واحدة من أهم المهام في مجال الروبوتات. يصف هذا البحث مشكلة تخطيط مسار للروبوتات متحركة استنادا الى خوارزميات الذكاء الاصطناعي المختلفة. ويجب ان يفي مسار الروبوت بعدد من المعايير ومنها طول المسار والسلامة. في هذا البحث يتم تطبيق ثلاثة طرق مختلفة وهي خوارزمية اليرقات المضيئة، خوارزمية سرب الطيور وخوارزمية اليراعات المضيئة وسرب الطيور المشوشة الهجينة المقترحة في بيئة ثابتة ومعروفة لحل مشكلة تخطيط المسارات وبثلاث حالات. في الحالة الأولى، روبوت متنقل واحد، وفي الحالة الثانية، ثلاثة روبوتات متنقلة مستقلة أما في الحالة الثالثة ثلاثة روبوتات متنقلة متتابعة. وتظهر صحة النموذج المستخدم للروبوت المتحرك وتوضح ان الخوارزمية الهجينة المقترحة تؤدي أداء أفضل من خوارزمية سرب الطيور وخوارزمية اليراعات المضيئة تحت نفس الشروط البيئية عن طريق الحصول على سرع سلسة وأقصر مسارات لعدد من الروبوتات المتنقلة.

---

*Corresponding author

Peer review under the responsibility of University of Baghdad.
https://doi.org/10.31026/j.eng.2019.06.07

Article received: 27/7/2018
Article accepted: 25/9/2018

الكلمات الرئيسية: الروبوت المحمول بعجلات، تخطيط المسار، بيئة ثابتة، خوارزمية اليراعات المضيئة، خوارزمية سرب الطيور.

1. INTRODUCTION

These days, robots have been applied in the many areas like in medical, military applications, space exploration, industrial and so on **Abbas, et al., 2016.** Path Planning was start in the middle of the 1960's and because of the computational time that is required in order to solve such problem rises dramatically while the size or dimension of the problem raises, this problem consider as an NP-hard (non-deterministic polynomial time). The aim of path planning is to plan an optimum path for wheeled mobile robot to navigate from its start point to its target point while shunning any obstacle that may located on its way consider a one of most essential task. So, according to this definition, path-planning problem is classified as an optimization problem **Han, 2007**. In any event, there are many paths for mobile robot to reach the goal, but actually, the superior path is adopted based on some optimization criteria such as least energy consuming, shortest distance or shortest distance and shortest time are most adopted criteria **Alam, et al., 2015**. Algorithms that used to address the problem of mobile robot path planning are divided into traditional algorithms such as (Road Map, Cell decomposition and Artificial Potential Field (APF) and into soft computing algorithms such as (Neural Networks (NNS), Genetic algorithms (GAs), Artificial Bee Colony (ABC), Particle Swarm Optimization (PSO) and Firefly algorithm (FF)). Obviously, each method has a number of advantages and disadvantages, which motivate researchers to treat powerful techniques **Mnubi, 2016**.

In this work, original particle swarm optimization, original firefly and proposed hybrid (FFCPSO) algorithms tin order to solve the Mobile Robot Path Planning problem are adopted.

The remainder part of this work is organized as follow: section2 perform the kinematic schematic for Wheeled Mobile Robot, section 3 perform the optimization methods (Chaotic PSO, Firefly and hybrid (FFCPSO)), section 4 perform the simulation results and paper conclusion is presented in section 5.

2. WHEELED MOBILE ROBOT SCHEMATIC

Fig. 1 show the model of non-holonomic wheeled mobile robot (NWMR) which is consists of right and left wheel for motion on the same axis and an omni-directional castor in face of cart in order to make mobile robot more stable **Al–Araji, 2014.** Each wheel has radius indicated by (R) and (W) indicates the distance between the left and right wheel, while the midpoint between the mobile robot wheels is indicates by (c).

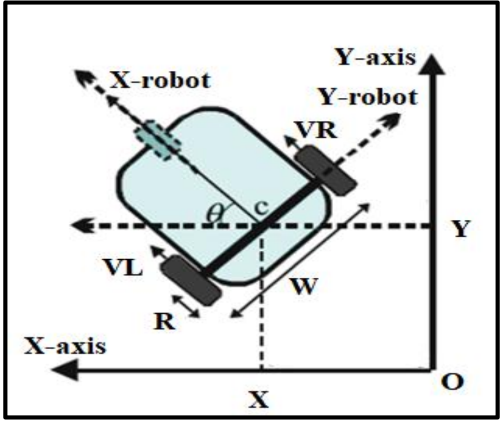

Figure 1. Mobile Robot platform **Al–Araji, 2014.**

Generally, the pose vector for non-holonomic wheeled mobile robot as in Eq. (1). While the global coordinate frame is defined as [X, O, Y].

$$\mathrm{S} = (\mathrm{x}, \mathrm{y}, \theta)^{\mathrm{T}} \tag{1}$$

Where (X, Y) are specified in the middle axis of wheels that act as the real position of (NWMR) while (θ) is act the orientation of (NWMR). Based on non-holonomic constraints as in Eq. (5) **Al–Araji, 2014**, the kinematic equations for (NWMR) can be represented as in Eq. (2), Eq. (3) and Eq. (4) **Araji, 2012** after provide the two statuses, the first status is a pure rolling wheel while the second status is without skidding wheels.

$$\dot{X}(t) = V_{lin}(t)\cos\Theta(t) \tag{2}$$

$$\dot{\mathrm{Y}}(t) = V_{lin}(t)\sin\Theta(t) \tag{3}$$

$$\dot{\Theta}(t) = V_{lin}(t) \tag{4}$$

$$-\dot{X}(t) sin\Theta(t) + \dot{Y}(t)\, cos\, \Theta(t) = 0 \tag{5}$$

Where, (V_{lin}) is denoted the linear velocity of platform while the platform angular velocity is denoted by (V_{lin}). Subsequently, reference linear velocity ($\mathcal{VVR}$) for the optimum route is calculated in Eq. (6) and the reference angular velocity ($\mathbb{WW}\mathcal{R}$) is calculated in Eq. (7) **Al–Araji, 2014.**

$$\mathcal{VVR} = \sqrt{(\dot{\mathrm{x}}\mathrm{rr})^2 + (\dot{\mathrm{y}}\mathrm{rr})^2} \tag{6}$$

$$\mathbb{WW}\mathcal{R} = \frac{\ddot{y}rr\ \dot{x}rr - \ddot{x}rr\ \dot{y}rr}{(\dot{x}rr)^2 + (\dot{y}rr)^2} \tag{7}$$

After that, the velocity of right wheel (VR) can be calculated as in Eq. (8) while the velocity of left wheel (VL) can be calculated as in Eq. (9) **Al–Araji, et al., 2011.**

$$VR = \mathcal{VVR} + \frac{W}{2}\mathbb{WW}\mathcal{R} \tag{8}$$

$$VL = \mathcal{VVR} - \frac{W}{2}\mathbb{WW}\mathcal{R} \tag{9}$$

Finally, the linear and angular velocities in terms of right and left wheels linear velocities can be calculated as in Eq. (10) and Eq. (11) **Al–Araji, et al., 2011.**

$$V_{lin}\,(t) = 0.5\,[\,Vl\,(t) + Vr\,(t)] \tag{10}$$

$$V_{ang}\,(t) = \frac{1}{W}\,[\,Vl\,(t) - Vr\,(t)] \tag{11}$$

3. OPTIMIZATION ALGORITHMS

In the next subsections, Chaotic PSO, Basic FF and proposed hybrid FFCPSO are used to locate the optimal control points (waypoints) within the interpolation points to find the optimum path from start to target points.

3.1 Firefly optimization Algorithm

Firefly (FF) algorithm is a population-based algorithm introduced Yang at Cambridge University algorithm tries to simulate the attraction behavior of fireflies and lighting in 2007. Firefly pattern. For simplicity, this algorithm based on only three rules **Yang, 2009**.

(1) All the number fireflies in the search space are the same gender so that any firefly can be attracted to other fireflies regardless of their gender.

(2) Their appealingness is relative to their luminousness, so for any couple of lighting fireflies, the less bright one will move towards the brighter one. If there are no brighter fireflies than appropriate firefly, it will move randomly.

(3) The luminousness of a firefly is determined by the cost function (light intensity) that need to be optimized.

Firefly algorithm consists of two steps; the first one is light intensity (F) while the second is attractiveness (β). The light intensity of each firefly is calculated using the Eq. (12).

$$F = Foe^{-\gamma rrij} \tag{12}$$

Where (Fo) be the maximal fluorescence strength of firefly and (γ) is the light observation coefficient and (rr_{ij}) is the distance between two fireflies. While the attractiveness is calculated as Eq. (13).

$$\beta = \beta_o e^{-\gamma rr^2} \tag{13}$$

The distance between firefly (i) and firefly (j) at (X_i, Y_i) and $\left(X_j, Y_j\right)$ can be calculated by the Eq. (14).

$$rr_{ij} = \sqrt{(X_i - X_j)^2 - (Y_i - Y_j)^2} \tag{14}$$

Thus, firefly (i) is start to move to brighter firefly (j) by Eq. (15):

$$X_i = X_i + \beta_o e^{-\gamma rrij^2}\left(X_j - X_i\right) + \alpha E \tag{15}$$

Where the first part in Eq. (15) gives the current position of the firefly, the second part is responsible for attractiveness while (α) is randomization parameter and (E) is vector of random variables, which makes the investigation of the search distance more effective. A firefly will be directed towards the brighter one, and if there is no brighter one surrounding to it, then it will move randomly as in Eq. (16).

$$X_i = X_i + \alpha(rand - \tfrac{1}{2}) \tag{16}$$

3.2 Chaotic Particle Swarm Optimization (CPSO) Algorithm

Although the PSO algorithm has the advantages of simple structure, easy to be describes and implemented, adjusts the less parameters, uses relatively small size of population, takes on fast convergence, good robustness and higher computational efficiency than the traditional method, it is easy to fall into local extreme value and cannot obtain the global optimal solution **Saud, et al., 2018.** In order to improve the ability of global searching and prevent a slide into the premature convergence to local minima, PSO and Chaotic map technique are combined to form a Chaotic Particle Swarm Optimization (CPSO) algorithm, which practically combines the behavior of chaotic searching with the population-based evolutionary searching ability **Liu Yi, 2016**. The logistic equation as in Eq. (17) **Hussain, et al., 2013**:

$$A^{t+1} = ⍰\, A^{t}(1 - A^{t}) \qquad (17)$$

Where (⍰) is the control parameter is equal to (4). The inertia weight factor (□) as in Eq. (18)

$$\square = \square f - [(\square \mathrm{f} - \square \mathrm{l})(t \backslash T_{max})] \qquad (18)$$

Where ($\square f$) is the maximum value if weight factor and ($\square l$) is the minimum value of weight factor, the new inertia weight ($\square_{new}$) as in Eq. (19).

$$\square_{\mathrm{new}} = \square * A^{t+1} \qquad (19)$$

The new update velocity is described in Eq. (20) and Eq. (21). In order to improve the global searching capability of standard PSO.

$$V_{ix}^{t+1} = \square_{new}\, V_i^t + c_1 r_1 (P_{besti} - X_i^t) + c_2 r_2 (G_{best} - X_i^t) \qquad (20)$$

$$V_{iY}^{t+1} = \square_{new}\, V_{iY}^t + c_1 r_1 (P_{besti} - Y_i^t) + c_2 r_2 (G_{best} - Y_i^t) \qquad (21)$$

Where c_1 is the personal learning factor, c_2 is the global learning factor, ($\mathrm{P}_{\mathrm{besti}}$) is the best weight of each particle and ($\mathrm{G}_{\mathrm{best}}$) is the best particle among all the particles in the population.

3.3 Hybrid (FFCPSO) proposed algorithm

Firefly (FF) algorithm is vastly used for solving optimization and engineering problems because only standard firefly applies for solving problem can produce superior results. Nevertheless, in the local search of firefly algorithm, small distance between fireflies may lead to random walk and delay in convergence. So in order to develop the firefly algorithm by increase convergence and avoidance it to fall into the local minimum, characteristics of chaotic PSO is mixed with in the FF algorithm to form hybrid optimization algorithm called (FFCPSO). The hybrid algorithm has the same procedure as the firefly approach with the exception that the position vector of (FF) algorithm can be written as follows:

$$D_{px} = \sqrt{\sum_{k=1}^{D} (P_{besti,k} - X_{i,k})^2} \qquad (22)$$

$$D_{py} = \sqrt{\sum_{k=1}^{D}(P_{besti,k} - Y_{i,k})^2} \tag{23}$$

$$D_{gx} = \sqrt{\sum_{k=1}^{D}(G_{best} - X_{i,k})^2} \tag{24}$$

$$D_{gy} = \sqrt{\sum_{k=1}^{D}(G_{best} - Y_{i,k})^2} \tag{25}$$

The position vector of the hybrid FF-CPSO algorithm as in Eq. (26) and Eq. (27)

$$X_i^{t+1} = \square_{new} X_i^t + c_1 * e^{-D_{px}{}^2}(P_{besti} - X_i^t) + c_2 * e^{-D_{gx}{}^2}(G_{best} - X_i^t) + \alpha E \tag{26}$$

$$Y_i^{t+1} = \square_{new} Y_i^t + c_1 * e^{-D_{py}{}^2}(P_{besti} - Y_i^t) + c_2 * e^{-D_{gy}{}^2}(G_{best} - Y_i^t) + \alpha E \tag{27}$$

The steps of proposed hybrid FFCPSO algorithm based path planning problem are described as follows:

Step 1: Create the mobile robot environment, which occupied by a number of static obstacles is represented by a circle shape with various size in 2-D workspace. The wheeled mobile robot is not a point; the dimension of the robot is added to the dimension of an obstacle to assuring the safety of a robot while trying in the environment.

Step 2: Generate the initial population of fireflies (pop) in the working environment randomly.

Step 3: Set firefly parameters (γ, β_o and α) and initialize P_{best} andG_{best}.

Step 4: The fireflies are estimated based on objective function. There are two estimation functions to imagine how they are relative to the optimal solution; the first one is minimum path length (ML) that make the wheeled NI- mobile robot can travel from start point to the target point with minimum travelling time as in Eq. (28).

$$ML = \sqrt{\sum_{i=1}^{n_p-1}(X(i) - X(i-1))^2 + (Y(i) - Y(i-1))^2} \tag{28}$$

While the collision avoidance (CA) is the second objective function that make the wheeled NI-mobile robot, can travel in the workspace safely by calculate distance between the mobile robot and static obstacle as in Eq. (29) and Eq. (30).

$$Dist(k) = \sqrt{\sum_{k=1}^{nobs}(X_p - X_{obs}(k))^2 + (Y_p - Y_{obs}(k))^2} \tag{29}$$

Where (X_p, Y_p) indicates to the interpolation points, (X_{obs}, Y_{obs}) is indicates position of static obstacle and K indicate the number of obstacles in our environment.

$$CA(k) = \begin{cases} 1 & if\ Dist(k) \le \mu \\ o & other\ wise \end{cases} \tag{30}$$

Where μ is denoted to minimum distance allowable between the path and static obstacle.

Step 5: during iteration loop, each firefly is updated by using Eq. (26) and Eq. (27) if the fitness of one firefly is great than another firefly or by using Eq. (15) if the fitness of two fireflies is equal.

Step 6: Exit if the maximum number of generation is satisfied, otherwise return to step four.

4. SIMULATION RESULTS

The dimensions of National Instrument (NI) wheeled mobile robot model: W=0.36 (meter), R=0.075(meter), L=0.40(meter) and sampling time is 0.1 (sec). The parameters setting for a hybrid algorithm as follows: The maximum number of iteration (T_{max}) is 80, the number of fireflies (pop) is 20, the flash absorption coefficient (γ) is 1, the randomness (α) is 0.2, the initial attractiveness (β_o) is 1, r_1 and $r_2$are random numbers between [0-1]; the social (c_1) and the cognitive (c_2) parameters are positive values equals 1.5. The maximum linear velocity of platform is 0.5 (m/s) and maximum angular velocity of platform is ± 2.77 (rad/sec). The simulation results is carried out in MATLAB package on a laptop (DELL) with processor type Intel(R) Core i7-7500 V CPU@ 2.70 GHz) and 8 GB RAM.

4.1: Case A

In this case, the start point for wheeled mobile robot is (100,100) and the target point is (900,900). The minimum distance based on hybrid FFCPSO algorithm (blue path) is (1147.5) cm at iteration (45), the minimum distance based on original FF algorithm (green path) is (1158.4) cm at iteration (78) and the minimum distance based on original PSO algorithm (red path) is (1148.8) cm at iteration (57) as show in **Fig. 2** and **Fig. 3**. Therefore, it is clearly to say that the proposed algorithm can find shortest path with less number of iteration than other algorithms. After executing the programs of various intelligent optimization algorithms ten times, the results for case A are summarized in **Table 1.** It is clearly to say that the proposed algorithm can provide optimum and more smoothness path than other presented algorithm in the case of single mobile robot.

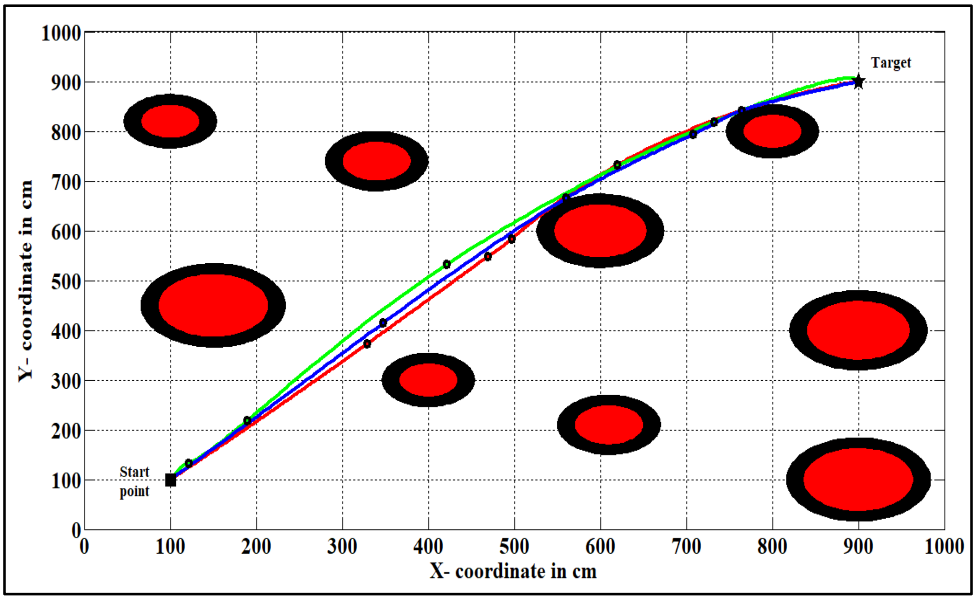

Figure 2. The shortest path for case A.

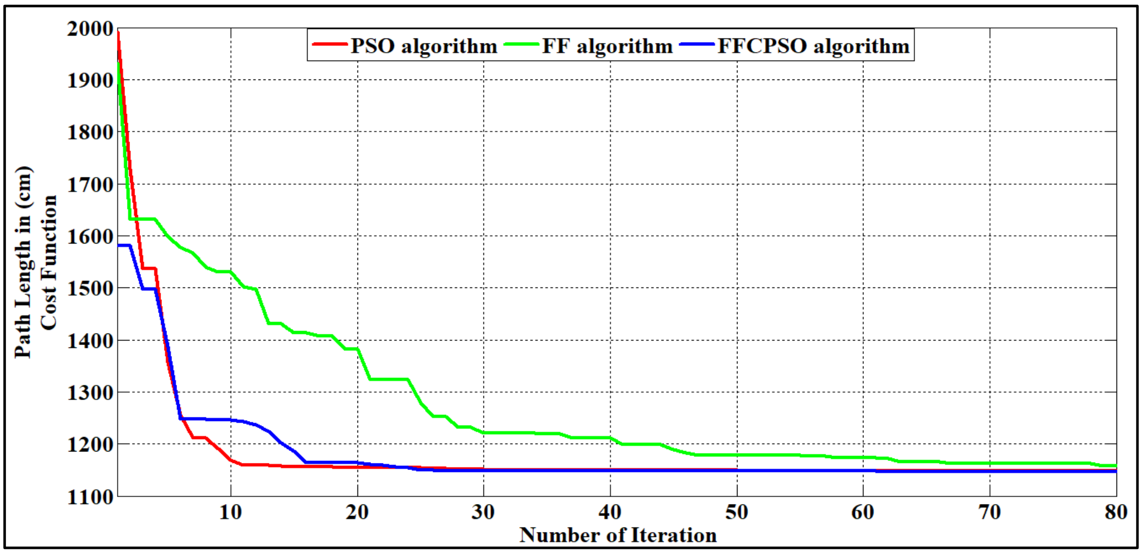

Figure 3. Variation of path length through iterations / case A.

Table 1. Comparison Results for case A.

Performance	PSO	Firefly (FF)	Hybrid (FF-CPSO)
Path Length in (cm)	1148.8	1158.4	1147.5
Iteration of best path	57	78	45
Travel time in (sec)	80	80	80

4.2: Case B

In this case, the start point for first wheeled mobile robot is (0,700), the start point for second wheeled mobile robot is (100,200) and the start point for third wheeled mobile robot is (500,100) while the target point for all robots is (900,900) as show in **Fig. 4** and **Fig. 5**. All the wheeled NI-mobile robots are start to move at (t=0) sec. After executing the programs of various intelligent optimization algorithms ten times, the results for case B are summarized in **Table 2.** The minimum distance based on hybrid FFCPSO algorithm for the first wheeled NI-mobile robot is (895.7) cm at iteration (33), the minimum distance for the second wheeled NI-mobile robot is (1072.2) cm at iteration (59) and the minimum distance for the third wheeled NI-mobile robot is (923.7) cm at iteration (44)**.**

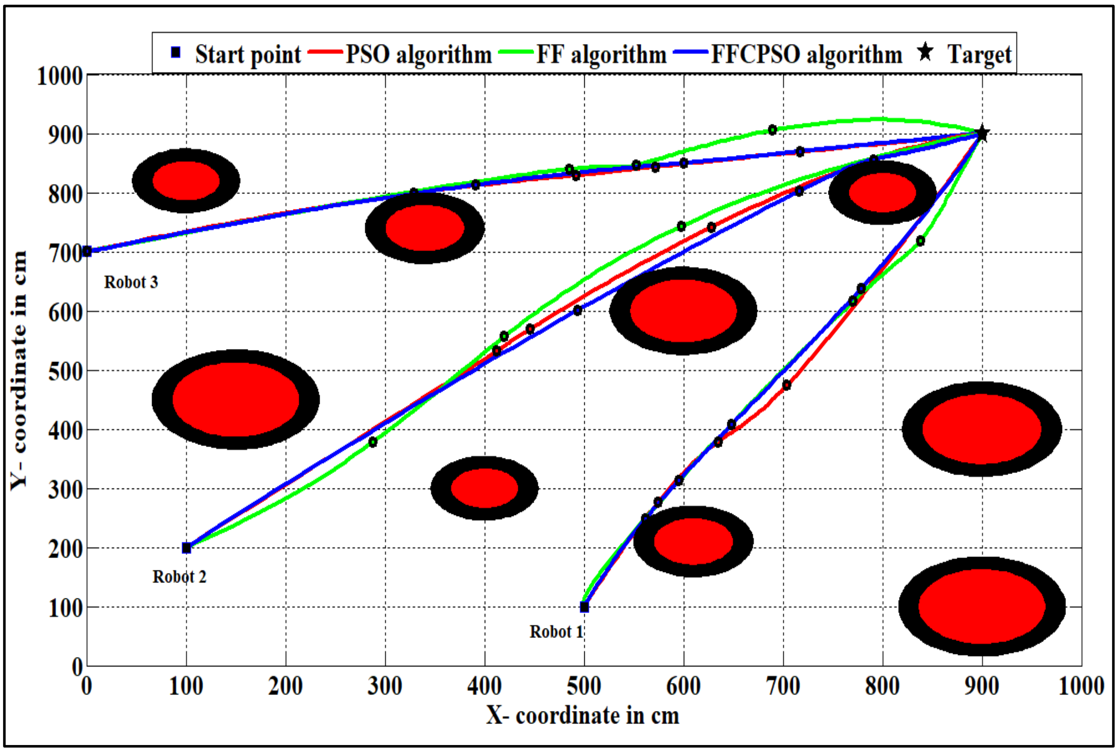

Figure 4. The shortest path for each robot/ case B.

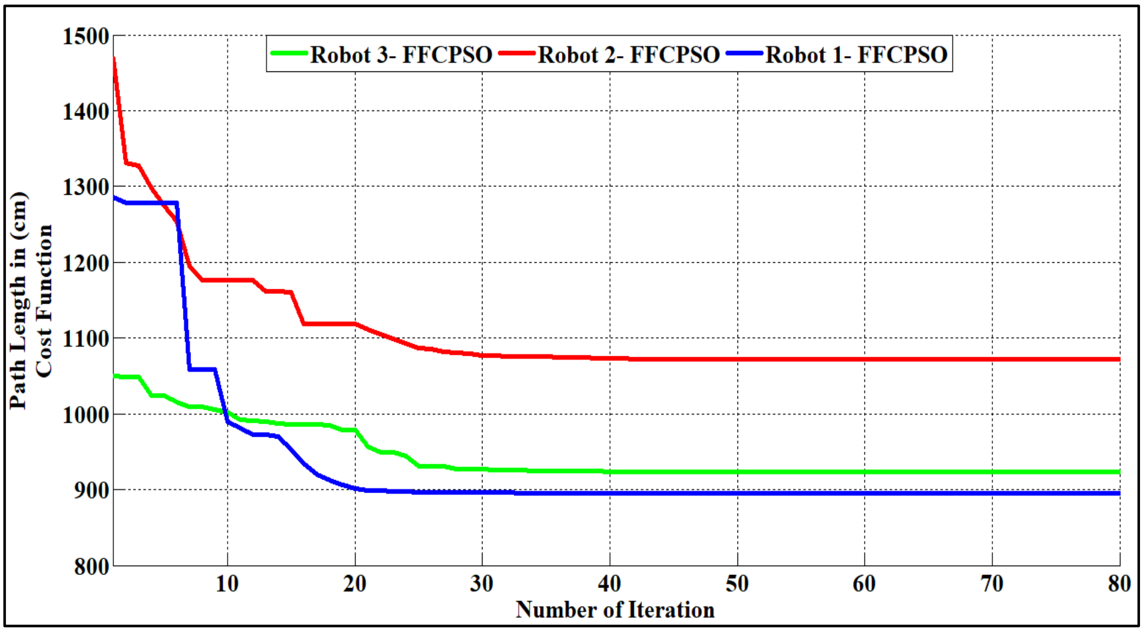

Figure 5. Variation of path length through iterations / case B.

Table 2. Comparison results for case B.

Robot No.	Performance	PSO	Firefly (FF)	Hybrid (FFCPSO)
1	Path Length in (cm)	897.7	900.2	895.7
	Iteration of best path	73	73	33
	Travel time in (sec)	40	40	40
2	Path Length in (cm)	1075.0	1082.5	1072
	Iteration of best path	68	44	59
	Travel time in (sec)	70	70	70
3	Path Length in (cm)	923.7	938.6	923.6
	Iteration of best path	58	77	44
	Travel time in (sec)	90	90	90

Then, based on Kinematic equations of wheeled NI- mobile robot, we can calculate the robot velocity for this case on its path. From **Fig. 6, Fig. 7** and **Fig. 8,** the linear velocity constrain of each wheel should not exceed 0.5 m/sec. In **Fig. 9, Fig. 10** and **Fig. 11**, explain the angular and linear velocity of platform, the angular velocity of platform should range between (-2.77, +2.77) rad/sec and the linear velocity constrain of platform should not exceed 0.5 m/sec.

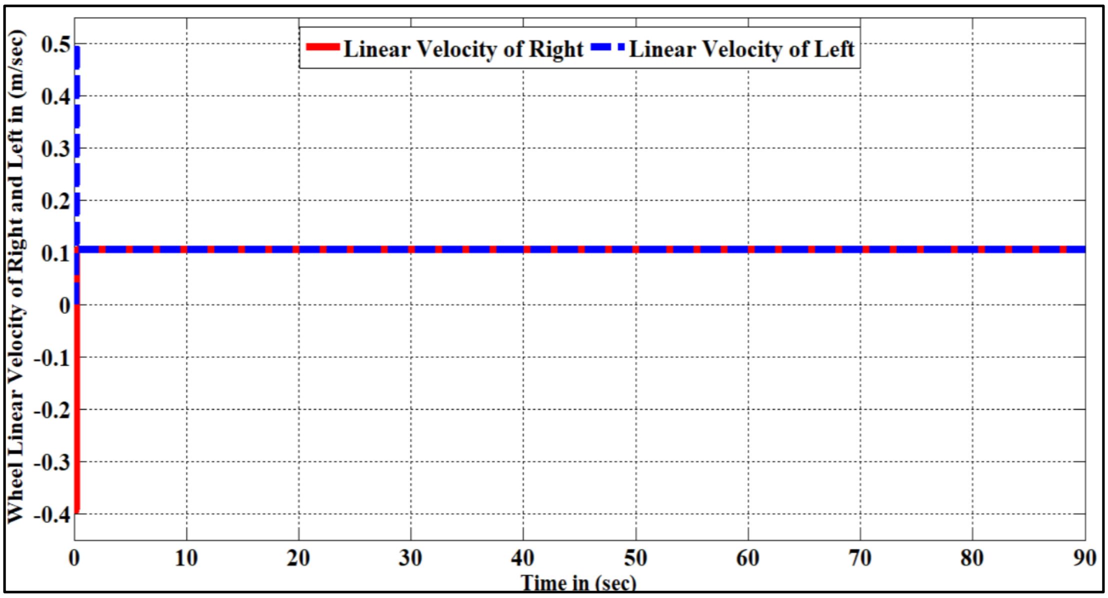

Figure 6. The wheel linear velocity of left and right actions for first mobile robot**.**

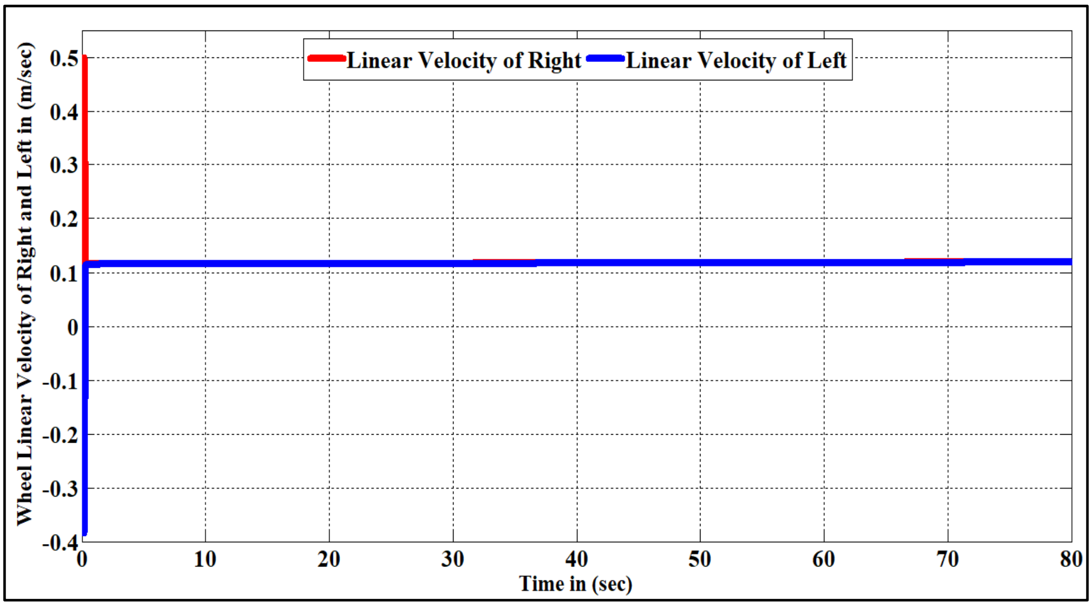

Figure 7. The wheel linear velocity of left and right actions for second mobile robot.

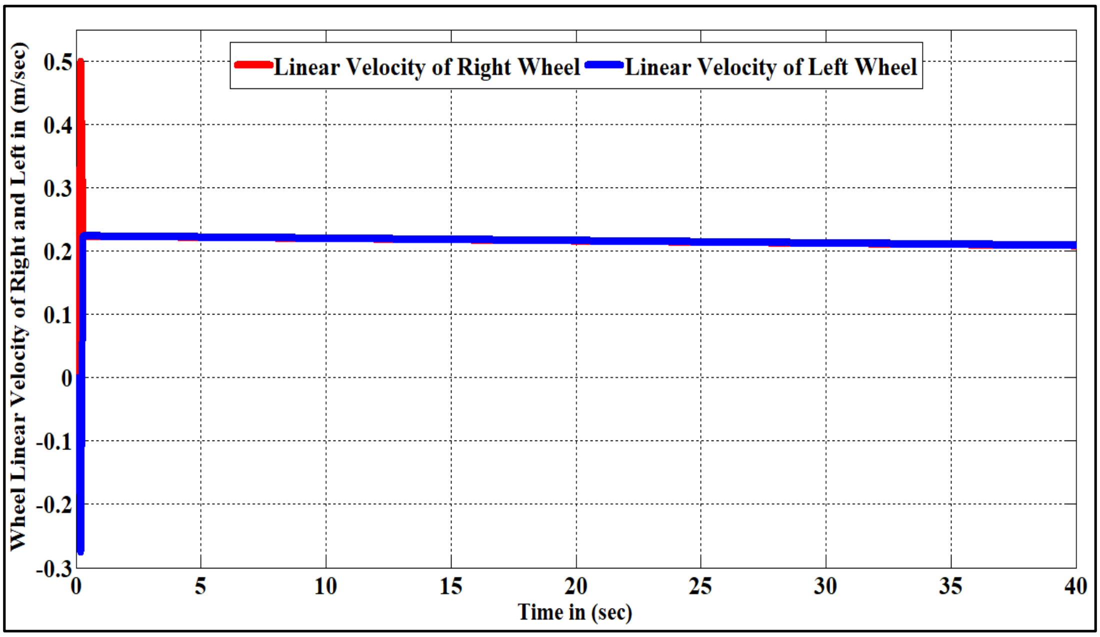

Figure 8. The wheel linear velocity of left and right actions for third mobile robot.

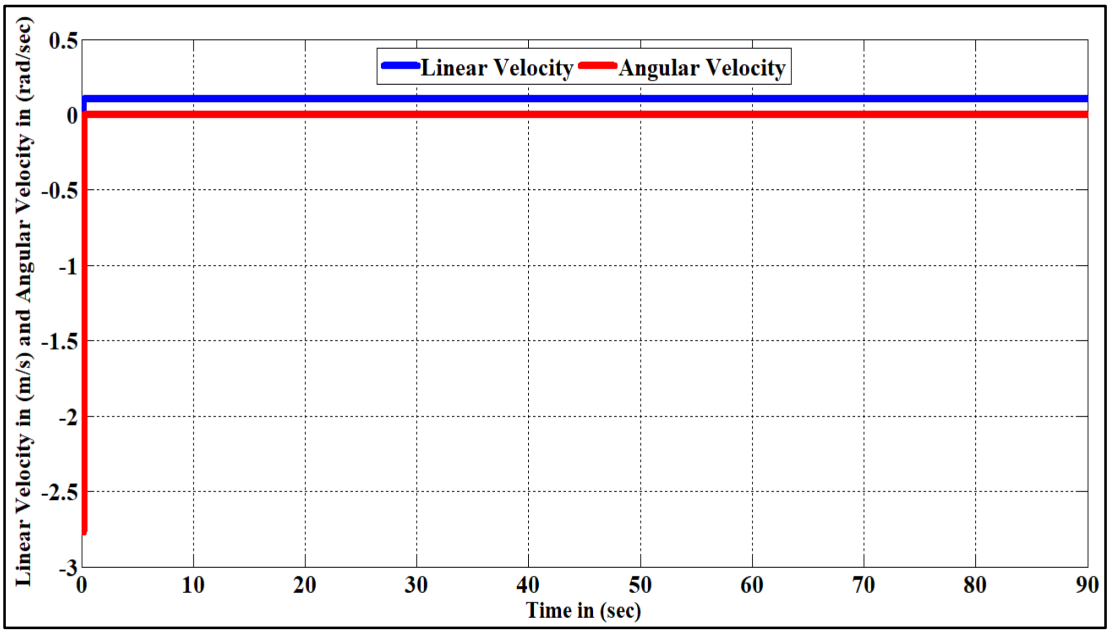

Figure 9. The platform angular and linear velocities actions for first mobile robot.

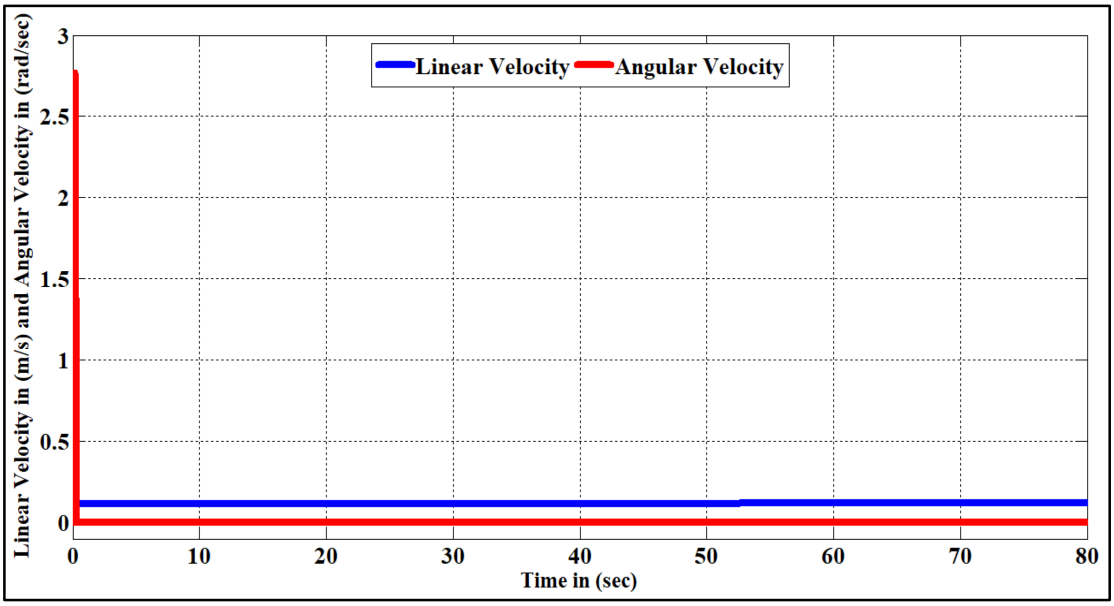

Figure 10. The platform angular and linear velocities actions for second mobile robot.

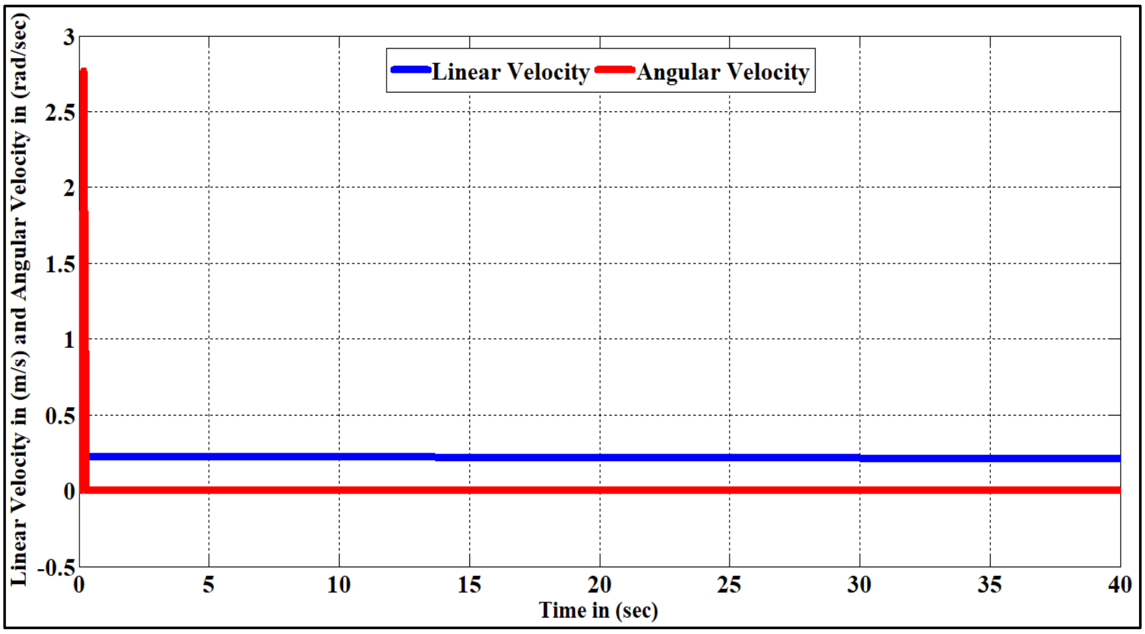

Figure 11. The platform angular and linear velocities actions for third mobile robot.

4.3: Case C

In this case, the start point for first wheeled mobile robot is (100,150), the start point for second wheeled mobile robot is (100,100) and the start point for third wheeled mobile robot is (100,50) while the target point for all robots is (900,900) as show in **Fig. 12** and **Fig. 13**. Based on wheeled-NI mobile robot length (I) and at (t=0) sec the first mobile robot is start to move toward the target, at (t=5) sec the second mobile robot is start to move while at (t=10) sec the third mobile robot is start to move. After executing the programs of various intelligent optimization algorithms ten times, the results for case B are summarized in **Table 3.** The minimum distance based on hybrid FFCPSO algorithm for the first wheeled NI-mobile robot is (1108.9) cm at iteration (48), the minimum distance for the second wheeled NI-mobile robot is (1147.9) cm at iteration (46) and the minimum distance for the third wheeled NI-mobile robot is (1187.1) cm at iteration (45).

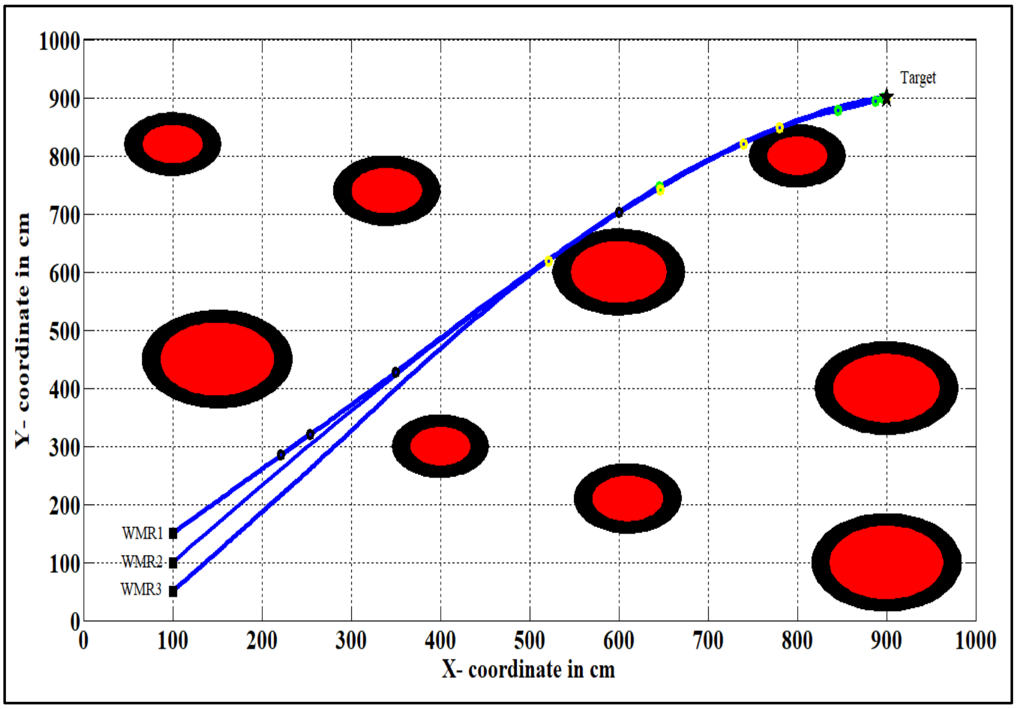

Figure 12. The shortest path for each robot / case C Based on (FFCPSO) method.

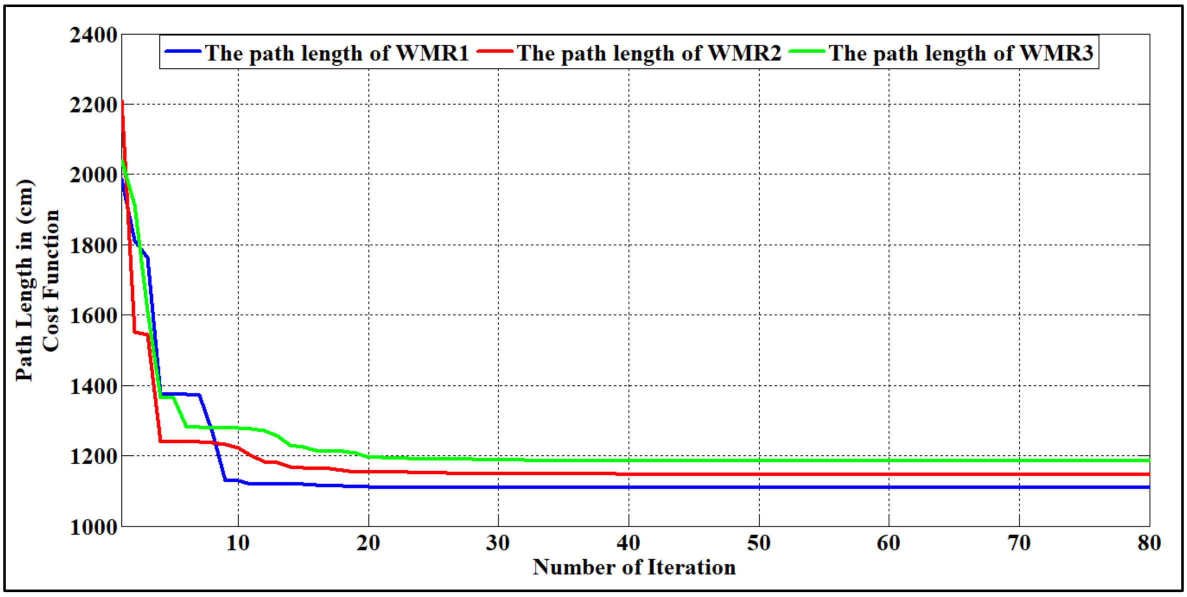

Figure 13. Variation of path length through iterations / case C Based on (FFCPSO) method.

Table 3. Comparison results for case C.

Robot No.	**Performance**	**PSO**	**Firefly (FF)**	**Hybrid (FFCPSO)**
1	Path Length in (cm)	1110.7	1113.7	1108.9
	Iteration of best path	47	64	48
	Travel time in (sec)	80	80	80

2	Path Length in (cm)	1148.4	1159.9	1147.9
	Iteration of best path	74	65	46
	Travel time in (sec)	80	80	80
3	Path Length in (cm)	1188.4	1194.27	1187.1
	Iteration of best path	74	53	45
	Travel time in (sec)	80	80	80

From **Fig. 14, Fig. 15** and **Fig. 16**, the linear velocity constrain of each wheel should not exceed 0.5 m/sec. In **Fig. 17, Fig. 18** and **Fig. 19**, explain the angular and linear velocity of platform, the angular velocity of platform should range between (-2.77, +2.77) rad/sec and the linear velocity constrain of platform should not exceed 0.5 m/sec.

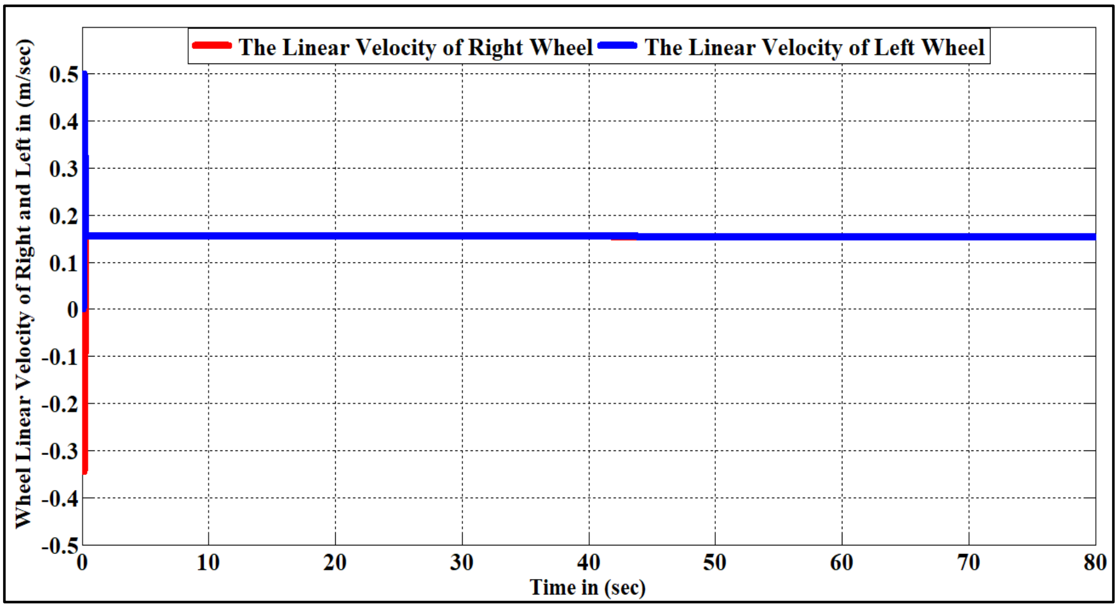

Figure 14. The wheel linear velocity of left and right actions for first mobile robot.

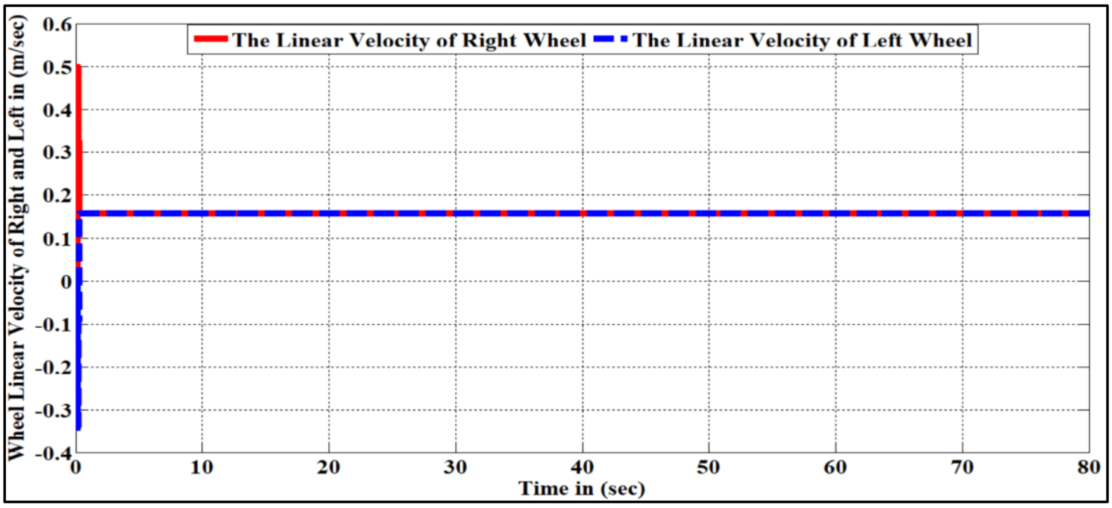

Figure 15. The wheel linear velocity of left and right actions for second mobile robot.

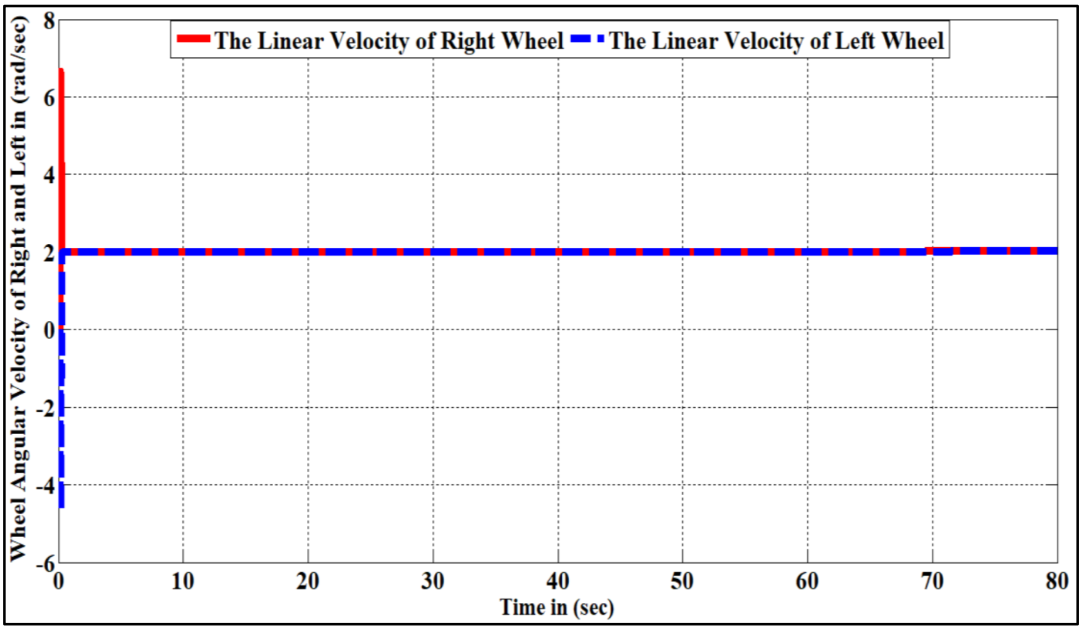

Figure 16. The wheel linear velocity of left and right actions for third mobile robot.

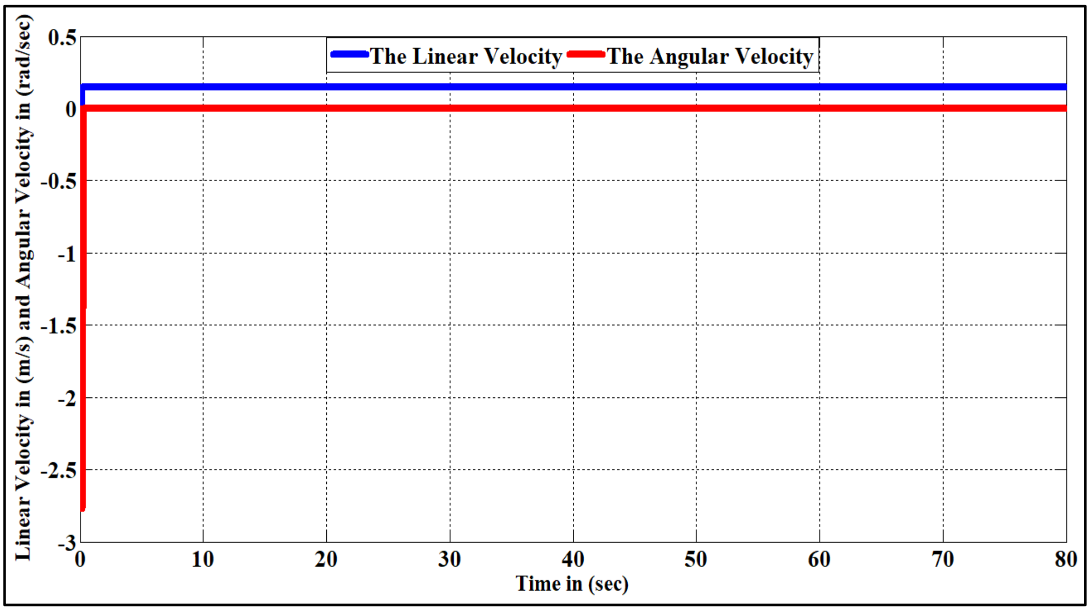

Figure 17. The platform angular and linear velocities actions for first mobile robot.

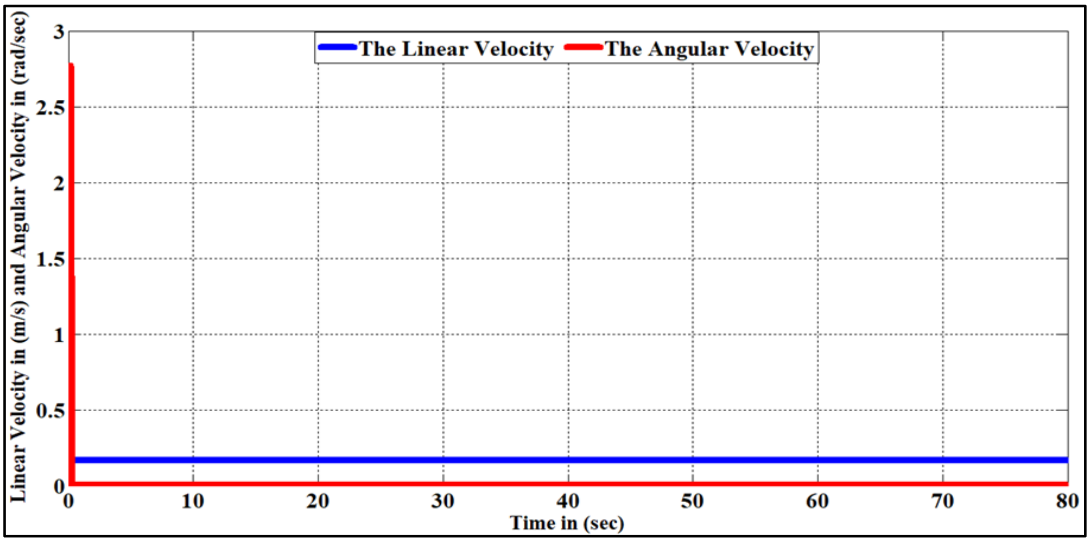

Figure 18. The platform angular and linear velocities actions for second mobile robot.

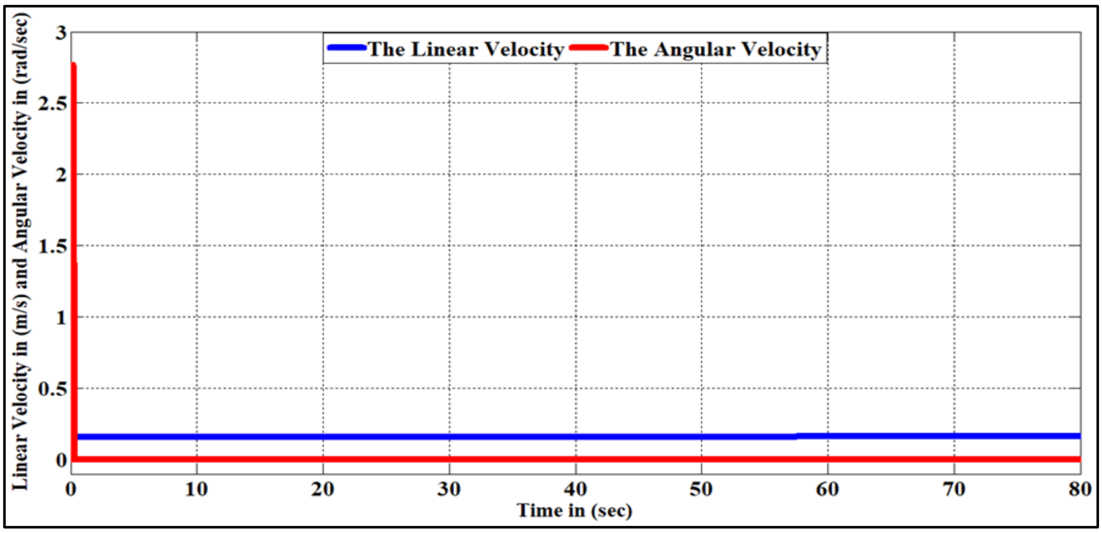

Figure 19. The platform angular and linear velocities actions for third mobile robot.

5. CONCLUSION

Path planning is an important part in robotic field that focus on find shortest path for mobile robot. There are number of optimization techniques used to solve this problem, PSO and FF algorithms are two of successful approaches in this application. In this paper, a hybrid A FFCPSO algorithm is proposed to find best route for mobile robots in three cases. From the simulation results of this study, show that the proposed hybrid optimization is able to find optimum path for multi robots better than original firefly and particle swarm optimization algorithms under the same environment conditions and can rightfully be regarded as a good choice due to its robustness and convergence speed in global and local search. In addition, the velocities actions are demonstrates the effectiveness of the optimization algorithms by showing its ability to produce smooth and small values of the angular and linear velocities of left and right wheels without sharp spikes this is lead to small power is wanted by the mobile robot to move on its path.

REFERENCES

- Abbas, N. H., J. A. Abdulsaheb**,** 2016, *An Adaptive Multi- Objective Particle Swarm Optimization Algorithm for Multi – Robot Path Planning***,** Journal of Engineering, No. 7,Vol. 22, pp.164-180.
- Alam, M. S., Rafique, M. U., and Khan, M. U., 2015, *Mobile Robot Path Planning in Static Environments using Particle Swarm Optimization,* International Journal of Computer Science and Electronics Engineering, Vol. 3, Issue. 3, pp. 253-257.
- Al–Araji, A. S., 2014, *Development of Kinematic Path –Tracking Controller Design for Real Mobile Robot via Back – Stepping Slice Genetic Robust Algorithm Technique*, Arabian Journal for Science and Engineering, Vol. 39, No. 12, pp.8825-8835.
- Al-Araji, A. S.,2014, *Design of On-Line Nonlinear Kinematic Trajectory Tracking Controller for Mobile Robot based on Optimal Back-Stepping Technique,* Iraqi Journal of Computers, Communication and Control & Systems Engineering, Vol. 14, No. 2, pp. 25-36.
- Al-Araji, A. S., Abbod, M. F. and Al-Raweshidy, H. S., 2011, *Design of an adaptive nonlinear PID controller for nonholonomic mobile robot based on posture identifier*, IEEE International on Control System, Computing and Engineering (ICCSCE), pp. 337-342.
- Al-Araji, A. S., Abbod, M. F., and Al-Raweshidy, H. S., 2011, *Neural autopilot predictive controller for nonholonomic wheeled mobile robot based on a pre-assigned posture identifier in the presence of disturbances*, The second International Conference on Control, Instrumentation and Automation (ICCIA), pp.326-331.
- Al-Araji, A. S., 2012, Design *of a Cognitive Neural Predictive Controller for Mobile Robot*, Ph.D. thesis, Electronic and Computer Eng. Dep., Brunel University, United Kingdom.
- Hussain, A. N., Melk, F., Rashid, M. A., Moamed, L., and Mohd Affendi, N. A., 2013, *Optimal Coordinated Design of Multiple Damping Controllers Based on PSS and UPFC Device to Improve Dynamic Stability in the Power System,* Hindawi Publishing Corporation Mathematical Problem in Engineering, pp.1-15.
- Saud, L. J., Hasan, A. F., 2018, Design of an Optimal Integral Back Stepping Controller for a Quad copter, Journal of Engineering, Vol. 24, No. 5, pp. 46-65
- Han, K. M., 2007, *Collision Free Path Planning Algorithms for Robot Navigation Problem,* Master Thesis, University of Missouri, Columbia.
- Liu Yi,” 2016, *Study on an improved PSO Algorithm and its Application for Solving Function Problem*, International Journal of Smart Home, Vol. 10, No. 3, pp.51-62.
- Mnubi, S. A., 2016, *Motion Planning and Trajectory for Wheeled Mobile Robot*, International Journal of Science and Research, Vol. 5 Issue 1, pp. 1064-1068.
- Yang, X., 2009, *Firefly algorithms for multimodal optimization*,” in: Stochastic Algorithms: Foundations and Applications, Lecture Notes in Computer Sciences, Vol. 5792, pp. 169-178.